Computer representation and animation of mathematical proofs with dynaboard

Marc Kaltenbach

UNITÉ DE RECHERCHE
INRIA-SOPHIA ANTIPOLIS

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél.:(1) 39 63 55 11

Rapports de Recherche

N° 700

COMPUTER REPRESENTATION AND ANIMATION OF MATHEMATICAL PROOFS WITH DYNABOARD

Marc KALTENBACH

JUILLET 1987

Computer Representation and Animation of Mathematical Proofs with Dynaboard

Représentation et animation de preuves mathématiques avec "Dynaboard"

Marc Kaltenbach†

Projet SIDE

INRIA - Centre de Sophia-Antipolis
Avenue Emile Hughes
06565 - VALBONNE Cedex - France

Abstract

The computer medium provides new ways of using both space and time in order to clarify the presentation to students of some complex deductive argumentations commonly found in Mathematics. We provide pedagogical justifications to using partially animated graphic network displays to facilitate student understanding of richly structured spatially organizable deductive information. We stress the active role given to the student in making the outer form of an argumentation evolve in relation to his understanding. Then we propose some functional features of a prototype system adapted to this form of information access. Finally we conjecture a probable evolution of visual idea processors toward greater decision autonomy and a greater symbiosis with human thought.

Résumé

Les écrans graphiques offrent des possibilités nouvelles pour la représentation de certains raisonnements déductifs complexes en mathématiques. Nous présentons des justifications d'ordre pédagogique en faveur d'une représentations spatiale du raisonnement capable d'évoluer en fonction des choix et préférences de l'étudiant. L'accent est mis sur le rôle actif de l'étudiant dans l'évolution de la représentation en accord avec l'état de compréhesion qu'il a atteint. Nous décrivons quelques fonctionalités de base d'un système adapté à cette forme d'accés à l'information. Pour terminer, nous situons nos dévelopements actuels dans le cadre d'une évolution probable des gestionnaires d'idées vers plus d'autonomie de décision et une meilleure symbiose avec la pensée humaine.

† On leave from Bishop's University
Faculty Box 59, Lennoxville, Quebec,
J1M 1Z7, CANADA.

Introduction

Objectives

Human-computer interfaces relying on graphics and direct-manipulation are fast becoming an essential part of computer based instructional technology. This paper considers their potential to improve the presentation to students of subject matters of significant structural complexity. Specifically, it describes our attempts to integrate space and time in order to clarify the presentation of complex deductive argumentations. The paradigm of navigation in a sea of information is applied to high-school and early university level Mathematics with the objective of creating an instructional medium intermediate between standard textbook presentations and the kind of personalized guidance provided by a human teacher.

Motivating context

Computer supported learning environments can be considered as one answer to the problem of scarce teacher time availability. This has provided the main motivation for the development of Computer Aided Learning and Computer Aided Instruction (O'Shea & Self, 1983). Efforts to combat the rigidities of menu driven, non-personalized systems are evidenced by recent proposals in favor of Intelligent Tutoring Systems (ITS) and Intelligent Computer Aided Instruction (ICAI) (Sleeman & Brown, 1982).

Another view of the computer medium is that it provides new and sometimes better ways to perceive complex human intellectual (abstract) creations. Personal experience, teaching tips from university colleagues and classroom experimentation have convinced the author that mathematical presentations often gain at being more visually organized than in textbooks. Various aspects of the teacher-blackboard-students interactions could be well approximated and sometimes improved by graphic representations the student can manipulate on a computer screen. To restrict on a very wide range of possibilities, we consider here only argumentations that admit spatial representations as graphs or similar structures we call "commented proof tree outlines" to be explored at various levels of detail. The argumentations are theorem proofs or solutions to exercises involving complex networks of deductions. The students will use our projected system, called Dynaboard, in association with the reading of a textbook to get alternative spatial and dynamic representations of complex proofs. He will make two-dimensional proof displays evolve in relation to the progress of his understanding. Insertion of Dynaboard within an ITS or its use as support for actual human teaching are other possibilities. For the time being, a distinction is made between the author of a Dynaboard "lesson" and the student of that lesson.

Early descriptions of ICAI systems usually started by emphasizing the autonomous decision facilities a system should have in order to regulate or participate in the unfolding of a "lesson". The design of an appropriate man-machine interface was considered afterward. This work proceeds in the reverse order. We first create display and exploration facilities for graph like structures and then progressively impart to the system increasingly higher levels of decision autonomy. Thus our long term objective is to create an ICAI system. This methodological choice springs from (current) difficulties in capturing human judgement into computer codes. We believe that for the near future, closer human-computer cooperation is likely to provide useful results not yet attainable by fully autonomous computer systems.

Past contributions to the design of the Dynaboard interface

The spatial (two-dimensional) display of information

Use of diagrammatic representations, emphasizing the relative placement of graphical objects to convey complex information, can be traced to the origins of civilization. However diagrams can be very cumbersome to manipulate and for this reason have often been abandoned for the more abstract forms of linguistic communication. The advent of sophisticated man-computer interfaces is giving them a new vitality by removing or attenuating many practical difficulties associated with the creation, manipulation and storage of complex images (Bennett, 1971).

Box and Arrow diagrams (B&A) have important practical uses. Methods for displaying graphs with nodes, corresponding to currently perceived units of information, and lines and arrows joining the boxes representing ways of linking these units together are common in Project Management (Apple, 1984a), Systems Analysis (Sage,1976), Semantic Networks (Winston,1979), to cite only a few among many applications. More closely related to educational purposes are the diagrams created for software visualization and animation (Cox & Pietrzykowski,1985), proof tree associated with Expert System search over Knowledge Bases (IntelliCorp™, 1985), and actual proof tree displays in Mathematics (Balacheff,1978). The rigidity of B&A diagrams can be attenuated by replacing the box formalism with more general graphic objects and a different notation for the links between objects. Links can be inferred by proximity grouping or more explicit associations like common enclosure, check marks or common "coloring".

Among the outstanding problems blocking a wider usage of network diagrams, extensions or analogs thereof, is the need to automate their drawing from abstract descriptions. Partial solutions are provided by Interpretive Structural Modelling (Warfield, 1967), and a variety of heuristics (Rowe & al.,1987). For the time being, excepted for very specialized and constrained problems such as VLSI design (Ayres, 1983), the general node placement problem has not received a fully satisfying solution. Approaches currently under study tend to combine formal results from Graph Theory with Knowledge Base ideas from Artificial Intelligence, (Tichy & Ward, 1987). Excepted for simple graph structures, semi-automatic procedures in which the computer provides a solution that is then corrected by the human operator seem unavoidable for the near future.

The temporal access to information

Ways of accessing information sequentially are backed by an extensive body of experience. Database systems (Chamberlin et al., 1976), allow efficient information access mechanisms adapted to a variety of user needs, from very discriminating query formulation systems to browsers, natural language interfaces and active help facilities (Michard, 1985). The problem of sequencing user controlled and context dependent accesses to complex information structures has received valuable solutions from Petri Nets Theory and applications to Process Control (Peterson, 1981), ergonomic studies of man-machine interfaces (Robertson, 1981), (Norman & Draper, 1986), and tutoring languages and systems, (Apple, 1984b) .

Synthesis : the merging of spatial and temporal characteristics

An important limit to the use of diagrams is that they quickly loose their appeal as they become larger and more complex. They must be simplified. A simplification may take the form of a segmentation into disjoint sub-diagrams; recourse to a hierarchy of increasingly abstract concepts is another possibility.

Integration of spatial and temporal dimensions in information retrieval is proposed as a check to the limitations that result when one of these dimensions is favored at the exclusion of the other. Visual displays are constrained by the size of the drawing area, delimited by the terminal video screen, and the care not to confuse the viewer with cluttered, information saturated, displays. Temporal access to information is constrained by the need to provide the user with adequate contextual information to help him steer a proper course among the multiple possible states of a complex information system. Proposals to improve the integration of spatial and temporal characteristics in information retrieval can be traced back to Nelson's hypernets (Nelson, 1970), in which a reader creates links between parts of a text to reorganize the information according to his personal needs. Methods adapted to "navigation" within linked page structures have been extensively studied in the ZOG system (Robertson & al., 1981). Weyer (1982), has proposed convincing pedagogical applications of the linking of text elements into dense networks.

Another related line of research has stressed the importance of whole-part organizations. They have become popular with the outline organizers proposed to facilitate text composition, (Apple, 1984c). With SADT Ross (1985) has pioneered the idea to give a greater importance to the spatial composition of individual displays in a hierarchical organization of information. More recent geographical database systems (Shneiderman, 1983), place less restrictions on the structure of individual displays; selection of one graphical object on a video screen leads to the display of its component parts, themselves expandable into further refinements, etc. It has also become possible to relax the strict hierarchical organization of displays and make individual displays less static. Objects on a single display can be functionally related; for instance pointing at one word within a text may serve to highlight all the occurrences of that word. Until recently, the creation and joining together of these functional features required extensive development work. Now this is made easier by the introduction of computer environments, such as the Design system (MetaSoftware, 1986), implemented on macintosh computers, offering a broad range of editing and navigation facilities as part of a classical programming environment. Efforts to make similar facilities directly accessible to a wider public by way of a simple but sufficiently powerful language are represented by the Boxer system of diSessa (diSessa, 1986). In summary a greater role is given to human visual information processing and to the computer capability to make complex pictures evolve in response to simple human input.

Outline of this report

First we provide justifications to using partially animated graphic network displays as one way to facilitate student understanding of richly structured spatially organizable deductive information. We stress the active role given to the student in making the outer form of an argumentation evolve in relation to his understanding. Then we describe some functional features of Dynaboard that are adapted to this form of information access. Finally we conjecture the probable evolution of visual idea processors toward greater autonomy of decision and better symbiosis with human thought.

Pedagogical display of mathematical structures

The structure of mathematical argumentations needs greater emphasis

There is a long history of attempts to narrow the distance between the "internal" form of an argumentation, reflected by multiple links between its component parts, and its actual display on a material medium. Medieval logicians saw great merits in characterizing and classifying diverse reasoning forms by diagrammatic structures (Brown, 1984). Modern logicians have proposed network-like notations. Thus Frege (1879), created a two-dimensional notation for predicate logic; Pierce (1897-1906), suggested that a few basic operations on network structures could constitute a viable alternative to predicate logic, a claim supported and extended in a recent book by Sowa, (1984). Spatial representations in the form of diagrams are also quite common in Mathematics as support for text descriptions.

The value of network diagrams, to outline the structure of argumentations, has not escaped pedagogues concerned with the teaching of mathematics and related domains of expertise characterized by a strong deductive component (Balacheff,1978). Indeed, in the mainly "linear" textbook presentations the judicious recourse to paragraphing and special use of verbal and typographic markers and connectors often proves insufficient to make clear the nonlinear structure of an argumentation. There are also problems associated with human short term memory limitations (Miller, 1956), which make it difficult holding together in mind the complex preconditions for the truth of a particular deduction.

A specific example will make the exposition more concrete. The network in fig.1 is our two-dimensional translation of a proof in Royden's book on Real Analysis (p.42) (Royden, 1968). So far, the figure may appear as classroom blackboard notes. We shall consider later how, in our opinion, the diagrams can be made independently readable, detailed or simplified, in response to student choices and preferences. Though not having yet a precise understanding of the particular semantics associated with the boxes and arrows and of their contents, the "reader" already gets at a glance an intuitive measure of the proof complexity. Sometimes other interesting properties of a proof are evident, just from the outer appearance of the graph display. For example, fig. 2 exhibits the property of symmetry. If the text version of a proof is read in association with consideration of its two-dimensional display, the gain is a faster relative placement of the reasoning parts in the reader's mind. In short, the spatial representation assists the reader in acquiring a global view of a proof and facilitates his assessment of the function and importance of particular components within the totality of the proof. Apart from enabling some form of pattern recognition on groupings of abstract entities like definitions, new terminology, symbols and thus easing memory recall, the major advantage of stressing whole-part relationships is to facilitate the formation of conceptual hierarchies. Spatial displays allows the simultaneous identification of various structural levels, from individual component boxes to cluster or sequences of boxes, etc. Recognition of these levels leads to simplifications and therefore to a temporal evolution of the display through the introduction of summary boxes giving a more abstract description of the steps of an argumentation. This, in turn, facilitates the classification of proofs, leading in the student's mind to a better integrated understanding of a whole subject matter.

Display of reasoning structure does not imply strict adherence to the axiomatic method

Various attempts have been made in the past to display theorem proofs in the form of directed graphs or "proof trees" (Balacheff, 1978). Usually in these graphs the nodes are boxes containing mathematical facts or hypotheses; with the arcs connecting the nodes are associated deduction rules. The graph, when it can be drawn practically, exhibits a complete chain of deductions from axioms to conclusions. This conception of the proper way of representing mathematical argumentations, when applicable, follows directly from the Axiomatic Method (Wilder, 1952), and suffers from the same criticisms (Kline, 1970; Thom, 1971, Davis & Anderson, 1979). Briefly stated, it is rarely the case that a mathematical proof includes all the (formal) steps from hypothesis to conclusion and no more. Formal proofs are together too rich and too poor. They are too rich since many detailed sequences of steps are really unnecessary; given some general instructions the reader is able to reconstruct the missing steps or is convinced he can do it. They are too poor because the emphasis on details masks the salient ideas. Isolated understanding of all the elements of complex chains of deductions may well force a student to accept a conclusion in a purely mechanical way; this acceptance does not imply understanding of general principles.

A formal proof is also poor for its lack of a pedagogical perspective. Information about a proof can be of significant help to students, to prevent mental blocks or simply to provide motivation. Rissland (1978) has documented this need for extra-logical knowledge in mathematical expositions. The major problem is to define a mixing of ingredients best adapted to individual student needs. Textbook authors are constantly faced with the necessity of a compromise between exposition of formal steps, references to formal steps, figures, examples, proof strategy statements and motivating facts like historical notes. The result is often a text aimed at a very specific audience. We suggest that a significant portion of mathematical text can be directly translated into computer supported diagrams that not only integrate the many facets of a mathematical argumentation but also avoid information overload by way of "on-line" adaptation of a presentation to the particular information requests of an individual student. The intended result is to make mathematical argumentations more easily accessible to a wider audience.

Student understanding activities need greater support

Reading mathematical text normally requires from the student to engage in mental reconstruction efforts obviously much greater than recreative reading. Usually they involve personal textbook annotations or extensive work with pencil and paper. Textbook presentations are fairly limited in providing support for this reconstruction process; it is assumed, often wrongly unfortunately, that the student possesses all the factual and procedural elements needed and the time available to achieve a comfortable level of understanding of what he reads. Computer supported mathematical presentations permit a significant relaxation of these constraints. Textbook personalization becomes possible; the pitfalls of axiomatic style presentations are avoided and mental work is assisted by hints and segmentation of difficult steps into sequences of sub-problems the student can solve by himself. Through editing facilities, the student can be given an active role in composing or adapting visual displays to his particular needs, a process called "self documentation" (O'Malley,1986). On the other hand details requested about a new concept definition may be integrated to the current display in an organic, non-disruptive way. Conversely details may be suppressed to reflect the evolution in student understanding, clarify the display and stimulate the search for further abstractions. The computer supported presentation style can also be better adapted to a student learning rhythm and visual momentum (Woods,1984).

In the absence of extensive experimentation, how much of the freedom given to a student in regulating the evolution of an argumentation (display) is compatible with the necessity to maintain logical completeness and consistency of display as well as observance of sound pedagogical principles is difficult to assess with precision at present. This issue of student freedom, or "non-intrusiveness" of computer decisions (Fisher & al., 1985), is closely related to that of capturing a student model (Rissland, 1984), that includes the student objectives, current state of knowledge, his pedagogical preferences etc., and using the model to guide the computer toward "intelligent" choices. Student modelling may be desirable; it may also prove to be extremely difficult, if not problematic to achieve satisfactorily (Sparck Jones,1986), in contexts as complex as the understanding of mathematical reasoning. Our working philosophy allows for extensive student freedom in directing the composition and evolution of computer displays. In counterpart the student has the possibility to switch on multiple automatisms that help avoid tedious repetitive manipulations and intervene in situations of (student) indecision; given a particular display state, the system in response to a user request can provide on-line guidance as to what to do and visualize next.

Freedom of choice may in part reduce the need for explicit student modelling. Consider for example the problem of computer system adaptation to student pedagogical preferences: direct manipulation of displayed objects can render superfluous an explicit statement in favor of a top-down presentation style, from general concepts to details, or a bottom-up presentation, or a shuttling back and forth between details and generalizations.

So far we have presented the idea of a mathematical proof display system in terms of general considerations of pedagogy. Before we shift toward the description of more concrete realizations a brief summary of consequent requirements is in order.

Summary : general requirements for Dynaboard

The proposed interface is intended to support two complementary ways of "understanding" a mathematical argumentation:

- It should permit global structure visualization at various levels of detail. When this is not directly possible, it should provide mechanisms to help the student reconstruct in his mind the global spatial organization of the argumentation.
- It should emphasize the understanding of whole-part relationships, ease the identification of useful abstractions and provide for alternative views of the subject-matter.

Concerning the modes of interaction with the student :

- Computer screen displays should include only the minimum amount of information that is consistent with the current needs of the student .
- Additional information should be readily available, preferably obtained through direct manipulations, and be inserted at a proper place on the display screen so that the whole display offers a well composed image.
- The student should have access to some editing facilities to re-organize a proof presentation according to his preferences.
- Assistance should be available to guide the student in his use of the system, to help resolve semantic ambiguities concerning the interpretation of particular objects drawn on the screen, plot his current position within the complex information structure, and assist him in the task of sequencing "reading" decisions.

System functional features

Commented proof tree outlines

We are interested by various ways of manipulating composite images of what we call "commented proof tree outlines". The term "graph" is also used for shorter reference. Fig. 1, 7, 8, 9 and 10 are simple but representative examples of these "graphs". Their component objects convey either mathematical information or "command" information to effect display state transitions. Network structures can be seen in which the nodes are rectangular boxes including semi-formal statements corresponding to diverse types of objects such as mathematical theorems, axioms, hypotheses, deduction rules, proof strategy statements, etc. The arcs linking the boxes may stand for causal relations or just indicate a preferred reading order. In addition there may be a variety of annotations, labels, geometrical figures, surrounded by invisible boxes.

Compared with the rigid formalism of proof trees there may be some ambiguity how to interpret these graphs. In the present context this should not be considered negatively. Usually the ambiguity can be removed by consideration of the context in which a graph has been obtained and the semantic of the individual objects involved. Otherwise, as we shall see later, the reactive property of screen displays can be used to insert at appropriate places the needed complement information. In summary, our treatment of mathematical proofs permits global visualization of argumentation structures; it leaves out minor details and avoids cluttering the display by not enforcing a rigid and cumbersome display "syntax".

In the following the use of these graphs is described from the perspective of a potential student user. Dynaboard overall command structure and editing facilities to assist an author enter a "lesson" in computer memory shall be documented in (Kaltenbach & Michard, 1987). Commented proof tree outlines are difficult to draw automatically and sufficiently fast. We describe various approaches by which a well composed commented proof tree outline can be rendered visible on the computer video screen and made to evolve in accordance with student information requests.

The temporal evolution of displays: the "collage" approach

We extend the desktop metaphor (Shneiderman, 1983), by the requirement that screen displays be well composed collages of stacked windows. We have created a small illustration of this idea, using the restricted set of end user facilities of Design (MetaSoftware, 1986), a semantic network processing system (Shapiro, 1975), running on macintosh computers. Design has full access to the macintosh user interface of windows and graphic functional features. In addition it allows one to link an extensive set of windows into a complex network structure. In any window, any rectangle (possibly including text or graphic objects) can be designated as a port to bring out the appearance of another pre-define window. With this, composite screen images of mathematical objects can be built and made to evolve incrementally in response to student.

An illustration with Design

The proposed sample exploration concerns a high school Geometry exercise (Balacheff, 1978). Figures 5 to 10 are screen copies intended to document a style of computer aided proof exploration made possible by Design. The actual implementation carries the explanation of concepts and deductive structures to greater depth; it includes 62 macintosh windows out of which 15 provide information assistance on how to interpret the displays and make them evolve. In the following, each figure is commented briefly in the order it appears in the exploration. In this presentation we suppose the student has already mastered the simple rules for using the system and fig. 5 corresponds to the screen display he his currently facing.

The problem statement in fig. 5 is distinguished from a standard book presentation by the inclusion of icons and of one thick border box. These objects have the property that a double mouse click (when the cursor is) over them brings out additional information in the form of new windows[1] opened, properly sized and positioned over the current display. It is possible to reverse the direction of an exploration by successively closing the currently "active" window. Short cuts are also provided by some icons.

Supposing the student has "clicked" inside the "proof" box, fig. 6 is obtained. This simple change in display composition is presented here only to illustrate a case of "implicit" advice. The new window location is such that the mouse cursor is placed over the "proof outline" box; a new double click will bring out the proof summary in fig. 7, which is the display the student is implicitly advised, but not compelled, to get. Of course the student could select the other choice and obtain fig. 10, but some extra work changing the cursor location would be required.

[1] To save space and get better print quality the top borders of the macintosh windows have been omitted from the figures.

Fig. 7 displays a digraph structure outlining the overall proof strategy. The arrows linking the boxes usually indicate a reading order. When it is wished to stress their interpretation as inferences or implications they are drawn with thicker lines. In addition an informal graph reading syntax specifies that boxes should be read from left to right and from top to bottom. Boxes with thick black borders include known facts. In this figure there are two windows for which additional information is available. Other "askable" objects are the lamp and the stop icons. The lamp icon contains the information that this property of parallelograms is often used to prove that three points are aligned. If the student needs no further detail to complete his understanding of the proof he can terminate the session by clicking on the stop icon.

The thick grey border box on the left is "askable" in two ways. A double click on the bold letter word "parallelogram" brings out the definition of a parallelogram as a four sided plane figure in which the opposite sides are parallel. This information is suppressed by closing the window containing it. Clicking elsewhere in the thick grey border box leads to a demonstration of the stated fact. Suppose the student has difficulty only with the other "askable" box. A click over it brings out fig. 8.

Fig. 8 introduces the idea that links between boxes can be expanded to provided liaison text to transform two dimensional displays into a linear form of discourse; it is also used to provide complement information on an inference that is made. The information is obtained by clicking on the "?" icons. We shall suppose the student has clicked on the "askable" box of fig. 8 to obtain fig. 9. Accepting the information in fig.9 the student may retrace his steps backward by closing the windows that are active or make direct jumps to particular displays linked to the icons.

Comments on this use of Design and more generally on the collage approach

On the positive side, whole-part relationships are explicit; the reader will note how the proof presentation can be progressively adapted to various information needs the student may have. A certain amount of freedom is left to the student in selecting depth-first or breadth-first exploration strategies or mixtures of these strategies. The transition of one display to the next often results only in a partial change of the whole screen image. This is valuable as we have found that even a very brief disappearance of the permanent part of an image followed by its re-drawing is quite disruptive. Thus, the incremental modification of screen displays does contribute to reduce the propensity of users "getting lost" in the information network (Canter et al., 1985). Practically, the collage approach is open to many further developments

On the negative side we note the exiguity of the display screen and the confusion created by the (non transparent) window borders. More seriously, there is a bias toward a top-down style of presentation. If a bottom-up style of presentation had been desired it would have been necessary to a-priori structure the proof accordingly. The mechanisms provided to navigate, i.e. effect transitions, in the information network are quite primitive. Experiments (with a few colleagues) confirmed the well known fact of users quickly losing track of where they are in the information network. Also, in our version of Design it was not possible to express constraints to insure the proper composition of displays. Infringement of the rules of navigation can result in confusion. Recovery procedures in the form of icon ports to various pre-specified orientation maps from which it is possible to branch out to a restricted number of key positions in the network structure entail tedious duplications of exploratory paths.

In the following we describe some of the functional features we have created to counteract some of the limitations encountered in this first experiment.

Needs for a complementary approach: the spatial viewing of large graphs

Facilities are provided to permit student exploration, interpretation and mental reconstruction of large commented proof tree outlines. Variations on the concepts of focusing, zooming and inspecting are proposed[2].

Focusing

For non trivial proofs the network structure to be drawn often occupies an area larger than the limits afforded by the view window in which it is to appear. The problem of how to help the user visualize a large commented proof tree outline, henceforth called the "world image", through a too small view window, has received various solutions in the past. We have found the scroll bar approach awkward to use when a large world image must be scrolled in any direction other than vertical or horizontal. The continuous displacement approach by dragging of an appropriate cursor symbol is very slow.

Our approach is by "focusing", a process by which a particular point of the world image is first selected and then made the center point of the view window. We provide alternative ways of

[2] The prototype is written in Ash-Prolog, [Michard & Monceyron, 1986], which is a linkage of C-Prolog, [Pereira, 1986], with the window facilites of The Brown Workstation Environment, [Pato & al., 1985], and runs on a sun/3 workstation.

selecting a point on the picture which we classify as either visual or verbal selections. Visual selection corresponds either to a pointer device click on one point of the world image appearing in the view window or to a click on a scaled down, "holophrasted" representation (Smith et al., 1984), of the world image (fig. 4)[3]. In these cases a sense of continuity between consecutive focus points can be created by way of a brief appearance of the translation vector as a thick arrow. Verbal selection is through keyboard entry of object names (or name substrings) fine tuned through menu selection if there are alternatives. New focus point selection is called indirect when an object, usually a box, is first selected through visual or verbal selection; afterward the precise new focus point is retrieved automatically as a predetermined attribute of the object. A menu selection among recent past focus points is also available. More sophisticated selection modes involving box contents are considered for a future implementation. For instance a query to locate various uses of a particular symbol (e.g. the symbol c in fig.1) could lead to an identification of their locations on the scaled down view of the world image.

So far in this description we have considered obtaining only one focus point. The presentation of a complex graph can also be animated by specifying sequences of focus points. These sequences can be pre-recorded, allowing a "cinema" mode of visiting the large image or they may be user specified. To create a customized sequence the user first sets on a recording mode; successive focus points are then recorded either automatically or after user confirmation, until the recording mode is switched off and the sequence obtained is given a name. The user obtains successive views of the world image corresponding to a particular menu selected sequence by successive mouse clicks. The sequence of recent focus points is also available and can be replayed either forward or backward in time.

zooming

The ability to continuously re-scale the view of a graph has multiple uses. Focusing on a point obtained from a scaled down view of the world image can be difficult if the components of the scaled down view are very small. For this reason the user can modify the size of the rectangle in which the scaled down view is made to fit. The world image can also be re-sized. We have installed a type of re-sizing we call "partial zooming" that leaves the boxes unchanged and adjusts only the relative positions of their center coordinates. To assist the user in selecting a

[3] Work is in progress to obtain the necessary interface between Ash-prolog, mathematical text editing and direct-manipulation CAD tools. The graph in fig. 4 is used currently to display the structure of the Ash-prolog graph management program.

scaling factor that results in no overlap of any box pair the smallest allowable scaling factor is displayed in a dialog box[4].

Inspecting boxes and icons

The user may wish to view a large part of the world image, implying that the box contents be summarized, and still be able to inspect the details of particular boxes. This is done by pointing, clicking and holding the locator device button while the cursor is on the target box. The box expanded view is then centered on the position of the box it replaces. Appropriate adjustments are made if there are interferences with the view window boundaries. When the device locator button is released the picture returns to its previous state. Several alternative expanded views may be available. This is used for instance to associate both vernacular and mathematical symbolic text with a box content. Choice of a particular alternative is obtained through the setting of a button[4]. To distinguish the boxes that have been inspected from those that have not, the user can specify that once a box has been inspected its border appearance be modified. A single (mouse) click is necessary to alternate between the focusing and the inspecting modes; this makes large graph explorations very fast. Sometimes it may be desirable to have simultaneous detailed view of the contents of several boxes. In our prototype the student reader is given the possibility to copy these views and position them on a "focusable" scratch pad that can be rendered visible or hidden on request.

Other facilities

Graph readability is strongly affected by the number and complexity of link connections between nodes. In fig. 1 for instance, additional links could have been drawn to explicit some relationships. One feature currently being added to Dynaboard will give the student the ability to alter the visibility of links. It will be possible to make selectively visible only the links that fulfill a particular function (e.g. emphasizing an inference). Alternative to explicit links, such as simultaneous highlight of related elements, and ways of facilitating references between nodes that are not simultaneously visible on the video screen are also to be implemented.

[4] Command and control objects (menus, buttons, dialog boxes, scaled-down graphs) are made visible when needed at fixed, but user modifiable, screen locations. In our prototype their are located by default in the left hand side of the display area (fig. 4).

Merging the spatial and temporal display facilities.

There is synergy between the two modes of presentation described so far. Temporal organization, implemented through the collage approach stresses whole-part relationships and the display of well organized, significant units but it is easy to get lost in the information network. With spatial representations the user can visualize display state transitions as a continuous process and is aided in mentally assimilating a world image but it is difficult to obtain fine grain levels of details without cluttering the display. Combining the two approaches is one way of compensating for the limitations of each approach. Our prototype includes a simple transfer mechanism from one mode of presentation to the other. Any window appearing in a collage can be attached to a specific focus point of the world image; the reverse transformation from world image to collage just reinstates the previous collage.

Solving a key problem with re-scaling or "attracting windows".

Collage and world image display facilities are useful but extreme solutions to the problem of viewing complex deductive information. An intermediate approach would provide smoother display transitions for situations where a two-dimensional view of a complex argumentation is appropriate; however it calls for the solution of a difficult practical problem: We seek efficient automated procedures to expand the view of any one node of a graph into that of a sub-graph properly inserted within the original graph and the converse capabilities to shrink a sub-graph into a single node. For arbitrary graphs this is already a hard problem. For commented proof tree outlines the difficulty appears much greater but the advantages would be significant. For instance students would be more free to adapt displays to their current state of knowledge and "lessons" would not be biased toward top-down or bottom-up styles of presentation.

Our approach in this direction is based on the idea that graphs need only appear to be re-restructured "on-line". Application is simple if shrinking transformations applied to a proof tree outline are such that the modifications, excluding connectors between nodes, are always positioned within the confines of the zone formerly occupied by the object(s) they replace. Then the converse operation of replacing an object by one of its extensions entails no on-line re-positioning of the entire display. There is however one difficulty posed by this approach. If numerous shrinking operations are performed on a graph, the distances between its various nodes may become very large in relation to the size of the view window. An example situation in fig. 3, shows the view window positioned only over a small fraction of the world image. This defeats the original goal of visualizing significant portions of a proof tree outline in order to extract information about its global structure !. The methodological principle we have adopted to solve this problem is to distinguish between the absolute placement of the nodes of a graph drawn on a limitless virtual canvass and their actual placement inside the view window. This "active screen" principle calls only for local reorganizations of a sub-part of the world image at the viewing time. Implementation is easy in the following case; if the distances between all nodes pairs are large, a re-scaling operation that leaves the sizes of the boxes (and their influence zones, if icons and labels are associated with the boxes) unchanged, permits larger parts of the graph to be visualized without distortion.

It is also possible for a re-scaling operation to be performed only for the part of the world image that will appear in the view window. However this operation may at times confuse the viewer as small changes in focus point may lead to rather large display changes. Then the student may have difficulty reconstructing in his mind a clear view of the world image. The distortion however may be acceptable if the student can also visualize a scaled down view of the world image and localize on it the part appearing in the view window (Billingsley, 1982). We have been satisfied with a preliminary evaluation of this approach implemented in C on a macintosh computer (Jean, 1986). The main problem with it is that it will be labor demanding to enter or modify the content of a "lesson".

A promising, but as yet hypothetical alternative to re-scaling, is sequential node placement allocation; the focused object appearing in the view window "attracts" its neighbors currently out of sight if there is room for them in the view window and position them according to a knowledge based "expert system" approach, (Winston, 1979).

Conclusion: steps toward intelligent browsers

So far we have been mostly interested by the problem of obtaining well composed dynamical displays of commented proof tree outlines. Many detailed developments and experimental evaluations of Dynaboard are still under way.

This work can also be placed in the wider perspective of developments at three levels. The first level concerns visual, passive representations. The second level deals with automatic decision facilities that do not involve computer formal processing of internal representations of the subject-matter and of a student model. The third level involves attempts to create and use these internal representations.

Level 1: diversifying the modes of visual representation

This report has concerned practical problems relative to the visualization of complex information derived from proof trees. There are however other computer based representational forms with the potential of removing some limitations of mathematical textbook presentations. Experiments with direct manipulations of visual analogies in the Physical Sciences (Woolf, 1987), can be extended to more abstract entities, as demonstrated in the Geometry Tutor (Anderson et al., 1985). Since appropriate concrete analogies are difficult to find for abstract forms of mathematical reasonings, the abstract symbolism must be directly dealt with. For instance, there is an important problem presenting complex nested definitions and highly recursive argumentations. Computer facilities should be developed to permit gradual transformations of mathematical statements into more familiar terminology and back. The student should be able to replace the terms of a definition by a specific example, should have access to prolog-like tracing facilities to check conformity of an instance to a general definition, etc. The need for many other representational facilities will probably result from attempts to build computer based discovery world covering an entire curriculum.

Level 2: expanding the set of automated functions

It appears quite certain that the increasing complexity of representational modes will entail the multiplication of functions that remain (at least partly) hidden to the student user. One example comes from the requirement that displays be consistent, i.e. relevant and non-contradictory. In the collage approach this can be insured by constraints that must be satisfied before a particular window can become visible. Another example concerns the internal coding of a subject-matter. Pictorial storage of a whole mathematical subject-matter would consume prohibitive amounts of computer memory space (unless some way is found of making use of video-disk technology[5]). On the other hand there is extensive redundancy among the various steps of an argumentation. A more compact way of storing the information is to parameterize reasoning pieces. Then a window appearing in a collage may have variables that are instantiated depending on the context in which a call is made for the window to become visible.

Level 3: toward greater human-computer symbiosis

Parameterized pieces of mathematical discourse are one step in the way of reducing the separation between "author" and "student" by giving the system abilities of autonomously

[5] Personal communication from David C. Littman.

filling gaps in an exposition. Considering recent advances in AI research the following situation is not inconceivable. A student is reading an article in the recent issue of a mathematical journal. The (still hypothetical) display system, provides a language (including direct manipulations) and ancillary facilities to enter in computer memory the information he reads as objects having both an external (visual) and an internal form. The system can verify the logical coherence of the information entered, provide explanations for possible gaps in an argumentation, refer to similar reasoning patterns considered in the past, suggest an ordered list of best candidate theorems (held in a data base) to help establish a particular result, evaluate student proposed solutions, etc. There is thus a continuous progression from systems that help clarify current knowledge to systems that actively participates to the development of new human knowledge.

References

ANDERSON, J.R., BOYLE C.F. & REISER, B.J. (1985). *Intelligent Tutoring Systems.* Science, 228, 456-462.

APPLE COMPUTER, (1984a). *MacProject.*
APPLE COMPUTER, (1984b). *Super Pilot.*
APPLE COMPUTER, (1984c). *Think Tank.*

AYRES, R. F., (1983). *VLSI silicon compilation and the art of automatic microchip design.* Prentice-Hall Inc., Englewood Cliffs.

BALACHEFF, N. (1978). *Les graphes de démonstration: outil pour l'étude des démonstrations naturelles.* Thèse 3ème cycle, Institut National Polytechnique de Grenoble.

BENNET, J.L. (1971). *Spatial concepts as an organizing principle for interactive bibliographic search.* In Walker, D.E., Ed., Interactive Bibliographic Search: The User/Computer Interface, 67-82. Montvale, New Jersey: AFIPS Press.

BILLINGSLEY, P.A.,(1982). *Navigation Through Hierarchical Menu Structures: Does it Help to have a Map ?.* Proc. of the Human Factor Society-26 Annual Meeting.

BOURNIQUE, R. & TREU S. (1985). *Specification and generation of variable, personalized graphical interfaces.* Int. J. Man-Machine Studies, 22, 663-684.

BROWN, M.A., (1984). *Generalized quantifiers and the square of opposition.* Notre Dame J. of Formal Logic, No. 4, 25; 17, 303-322.

CANTER D., RIVERS R. & STORRS G. (1985). *Characterizing user navigation through complex data structures.* Behaviour and Information Technology, 4, 2, 93-102.

CHAMBERLIN, D.D., ASTRAHAM, M.M., ESWAREN, K.P., GRIFFITHS, P.Q., LORIE, R.A., MEHL, J.W., REISMAN,P. & WADE, B.W., (1976). *SEQUEL2: A unified approach to data definition, manipulation and control.* IBM J. of System and Development, 20, 560-575.

CONNOR, M.F., (1980), *SADT: structured analysis and design techniques introduction,* Engineering Conference Records. New York, IEEE press, 138-143.

COX P.T.& PIETRZYKOWSKI T., (1985). *LOGRAPH: a graphical programming language.* Proceedings, Compint, Montreal.

DAVIS P. J. & ANDERSON J. A. (1979). *Nonanalytic aspects of mathematics and their implication for research and education.* Siam Review, 21, 1, 112-127.

DEDE, C. (1986). *A review and synthesis of recent research in intelligent computer-assisted instruction.* Int.J. Man-Machines Studies, 24, 329-353.

DISESSA, A. A., (1986). *Notes on the Future of Programming: Breaking the Utility Barrier.* In User Centered System Design, Norman & Draper ed. LEA.

EGAN D.E. & SCHWARTZ B. J., (1979). *Chunking in recall of symbolic drawings.* Memory & Cognition 7 (2) 149-158.

FERGUSON, E. S., (1977). *The Mind's Eye: Nonverbal Thought in Technology.* Science, 197,4306, 827-836.

FISCHER, G., LEMKE, A. & SCHWAB, T. (1985). *Knowledge-based help systems.* In L. Borman & B. Curtis (Eds). *Human Factors in Computing Systems-II,* CHI'85. ACM, San Francisco, 161-167.

FREGE G. *Begriffsschrrist.* Translated by van Heijenenoort, 1967.

INTELLICORP (1985). KEE™ : System Development Software, User's Manual Version 2.0.

JEAN, M., (1986). *Un interface pour l'exploration d'une preuve mathématique.* Third year student project, Dept. IRO, University of Montreal.

KALTENBACH, M. & MICHARD, A. (forthcoming). *Des outils Ash-Prolog pour la gestion de graphes.* Rapport technique INRIA.

KLINE, M., (1970), *Logic versus pedagogy.* American Mathematical Monthly, 77, 264-282.

METASOFTWARE CORPORATION. (1986). *Design™*, Boston, Mass.

MICHARD A. (1985). *Reconnaissance et génération de plan d'actions: application à la réalisation de systèmes auto-explicatifs.* Rapport INRIA 382.

MICHARD, A. & MONCEYRON, E., (1986). *Le système graphique Ash-Prolog et son utilisation pour le prototypage rapide d'interfaces hommes-machine.* Rapport technique INRIA 76.

MILLER, G.A., (1956). *The Magical Number Seven plus or minus two: Some limits on our capacity for processing information.* Psychological Review, 63,81-97.

NELSON, T.H. (1970). *No more teacher's dirty looks.* Computer Decisions. 9(8): 16-23.

NORMAN, D.A. & DRAPER, S.W. ed. (1986). *User centered system design.* LEA.

O'MALLEY C.,(1986). *Helping Users Help Themselves.* In User Centered System Design, Norman & Draper ed., LEA.

O'SHEA, T., SELF, J. (1983). *Learning and teaching with computers.* Engelwood Cliffs: Prentice Hall.

PATO, J.N., REISS S.P. & BROWN M.H., (1985). *The Brown Workstation Environment: Programmer's Manual, version 2.0.* Department of Computer Science, Brown University.

PEIRCE, C.S., (1960). *Collected Papers of Charles Sanders Peirce.* Edited by A.W. Burks, Harvard University Press.

PEREIRA, F. (ed.), (1986). *C-Prolog User's Manual, version 1.5.* EdCAAD, University of Edinburgh.

PETERSON, J.L.,(1981). *Petri Net Theory and the Modeling of Systems.* Prentice-Hall, Englewood Cliffs, N.J.

RAEDER, G. A (1985). *Survey of Current Graphical Programming Techniques.* Computer.

RICH, E., (1983). *Users are individuals: individualizing user models.* Int. J. Man-Machine Studies, 18, 199-214.

RISSLAND MICHENER E. L., (1978). *Understanding Understanding Mathematics.* Cognitive Science, 2, 361-383.

RISSLAND, E. L. (1984). *Ingredients of Intelligent User Interfaces.* Int. J. Man-Machines Studies, 21, 377-388.

ROBERTSON, G., McCRACKEN & NEWELL, A. (1981). *The ZOG approach to man-machine communication.* Int. J. Man-Machine Studies. 14,461-488.

ROSS, D.T. (1985). *Applications and Extensions of SADT.* Computer ,18,4.

ROWE, L.,DAVIS, M.,MESSINGER, E., MEYER, C.,SPIRAKIS, C. & TUAN A., (1987). A *Browser for Directed Graphs.* Software-Practice and Experience, 17(1), 61-76.

RUMELHART, D. E., (1980). *Schemata: The Building Blocks of Cognition.* In Theoretical Issues in Reading Comprehension, (Rand J. et al. eds.), LEA, 33-58.

SAGE, A.P. (1976). *Methodology for Large Scale Systems.* Academic Press.

SHNEIDERMAN B. (1983), *Direct Manipulations: a Step Beyond Programming Languages.* Computer, 57-69.

SLEEMAN D.I. & BROWN J.S. ed., (1982). *Intelligent tutoring systems.* Academic Press.

SMITH, S.R., BARNARD, D.T. & MACLEOD, I.A., (1984). *Holophrasted displays in an interactive environment.* Int. J. Man-Machines Studies 20, 343-355.

SOWA, J.F. (1984). *Conceptual structures : information processing in mind and machine.* Addison-Wesley.

SPARCK JONES K., (1986). *Issues in User Modelling For Expert Systems.* In Artificial Intelligence and its Applications, edited by A.G. Cohn and J.R. Thomas, John Wiley & Sons.

THOM, R. (1971). *"Modern" Mathematics: An Educational and Philosophic Error?.* American Scientist, 59, 695-699.

TICHY, W.F. & WARD B., (1987), *A Knowledge-Based Graphical Editor.* Technical report 3/87, University of Karlruhe.

WARFIELD, J., (1967). *Societal Models.* AP.

WEYER, S.A. (1982). *Searching for information in a dynamic book.* Xerox Corporation, P82-00016.

WILDER R.L. (1952). *An introduction to the foundations of Mathematics.* John Wiley & Sons.

WINSTON, P.H. (1979). *Artificial Intelligence.* Addison-Wesley.

WOODS, D., D., (1984). *Visual momentum: a concept to improve the cognitive coupling of person and computer.* Int. J. Man-Machine Studies, 21, 229-244.

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique